Remarks

In view of the following remarks, Applicant respectfully requests that the rejections be withdrawn and the application be forwarded along to issuance. Claims 1-26 are pending. Support for the amendment to claim 24 may be found in claim 13 and throughout the application. Entry of the amendment is request as the amendment clarifies issues for potential appeal.

Statement of Substance of Examiner Interview dated 8/7/08

Applicant would like to thank Examiner Miranda Le for her time in discussing this application over the phone on 8/7/08, with Applicant's attorney Nathan Grebasch.

During this interview, the §103 rejections based on Goward and Viswanath were discussed. The substance of the interview is included in the following discussion. The Examiner initially indicated that the claims may be allowable based on the arguments provided below. The Examiner additionally indicated that the claims may be allowable if an amendment were to be entered that differentiated the claims from Java implementations. No Agreement was reached.

If any issues remain that would prevent the allowance of the application, Applicant requests that the Examiner contact the undersigned attorney to resolve the issues.

Claim Objections

Claim 21 is pending an objection as failing to include "a processor or memory." Applicant takes the Examiner's comments to be an objection related to the inclusion of hardware for storing/executing software aspects. Applicant traverses the rejection. In particular, claim 21 recites "means for executing". Thus, claim 21 includes means that are capable of executing "at least a portion of the plurality of byte codes". Removal of the pending rejection is requested and allowance is solicited.

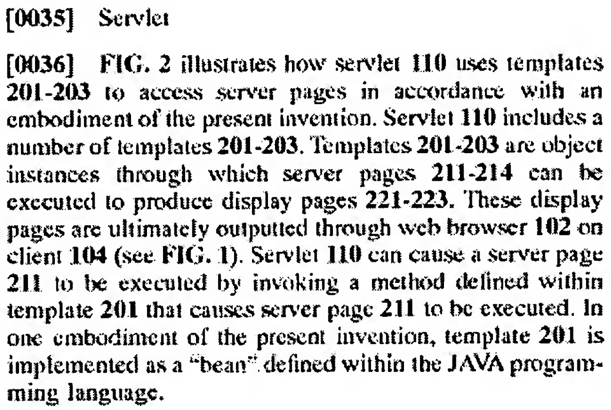
35 U.S.C. §103 Rejections

Claims 1-26 stand rejected under 35 U.S.C. §103(a) as being unpatentable over U.S. Patent Application Publication No. 20020120677 ("Goward") in view of U.S. Patent Application Publication No. 20030125966 ("Viswanath"). Applicant respectfully disagrees.

Claim 1, in part recites:

- receiving a request for a Web page;
- identifying an Active Server Page associated with the requested Web page, wherein the Active Server Page includes a compiled user interface template created using an Active Server Page Language which when compiled is executed through an application programming interface developed using a system language to generate the requested Web Page in the system language from the user interface template created using the Active Server Page Language;
- executing the Active Server Page through the application programming interface to generate the requested Web page; and
- providing the requested Web page to a source of the request.

Presuming, for the sake of argument only, that the combination of Goward/Viswanath is proper, the combination of Goward/Viswanath still fails to teach or suggest each and every feature as recited in claim 1. For example, consider the feature of "identifying an Active Server Page associated with the requested Web page, wherein the Active Server Page includes a compiled user interface template". In making out the pending rejection, the Office incorrectly relies on Goward paragraph [0036] as teaching this feature. For the Office's convenience, the section relied on by the Office is reproduced below.



[0035] Servlet

[0036] FIG. 2 illustrates how servlet 110 uses templates 201-203 to access server pages in accordance with an embodiment of the present invention. Servlet 110 includes a number of templates 201-203. Templates 201-203 are object instances through which server pages 211-214 can be executed to produce display pages 221-223. These display pages are ultimately outputted through web browser 102 on client 104 (see FIG. 1). Servlet 110 can cause a server page 211 to be executed by invoking a method defined within template 201 that causes server page 211 to be executed. In one embodiment of the present invention, template 201 is implemented as a "bean" defined within the JAVA programming language.

Goward, paragraph [0036].

Goward fails to teach the above feature as the template in Goward is merely used by the servlet <u>to access the server page</u> (e.g., an active server page). Goward, paragraph [0019]. Thus, instead of teaching "the Active Server Page including a compiled user interface template created using an Active Server Page Language" as is recited in claim 1, Goward is limited to teaching that a template can be used to access the server page, such as the active server page. Goward, paragraph [0019]. Goward's failure to teach that "the Active Server Page includes a compiled user interface template" may also be seen in Goward Figure 2. Goward Figure 2 clearly that shows that the template is included in the servlet and not in an active server page.

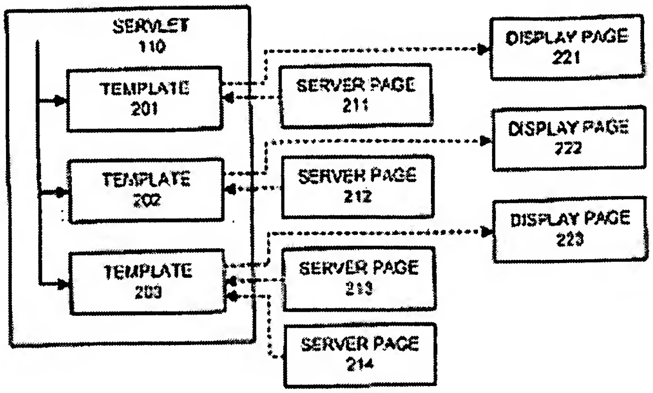


FIG. 2

Goward, Figure 2.

Moreover, Goward fails to teach or suggest the feature of "an application programming interface developed using a system language to generate the requested Web Page in the system language from the user interface template created using the Active Server Page Language" because Goward never teaches that the requested Web page may be generated "from the user interface template created using the Active Server Page Language". Instead, Goward simply teaches that the servlet can use the templates 201-203 to access server pages. Goward, paragraph [0036].

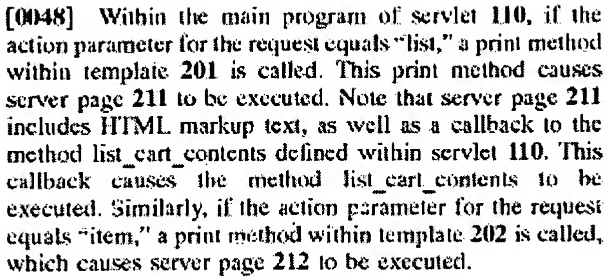
In addition, the Office failed to cite any portion of Goward as teaching the feature of a "user interface template created using the Active Server Page Language". The pending rejection merely contends that the secondary reference, Viswanath, teaches "Active Server Page Language" without explaining why one of ordinary skill in the art at the time of the invention would have know to apply the contended Viswanath teachings to Goward's templates. Absent some showing that Viswanath teaches a user interface template created in the above manner, the rejection is improper.

Additionally, the Office has failed to establish how the teaching of a template in Goward teaches or suggests a <u>user interface template</u> as recited in claim 1.

For at least the above reasons, a *prima facie* case of obviousness under 35 U.S.C. §103(a) has not been shown. Removal of the pending rejection is respectfully requested and allowance is earnestly solicited.

Claims 2-5 depend from claim 1 and are allowable as depending from an allowable base claim. These claims are also allowable for their own recited features which are neither taught nor suggested by the combination of Goward/Viswanath.

In particular, the rejection to claim 3 is improper as Goward paragraph [0048] cited as teaching the feature of "wherein the user interface template contains HTML code" fails to do so. For the Office's convenience, Goward paragraph [0048] is reproduced below.



[0048] Within the main program of servlet 110, if the action parameter for the request equals "list," a print method within template 201 is called. This print method causes server page 211 to be executed. Note that server page 211 includes HTML markup text, as well as a callback to the method list_cart_contents defined within servlet 110. This callback causes the method list_cart_contents to be executed. Similarly, if the action parameter for the request equals "item," a print method within template 202 is called, which causes server page 212 to be executed.

Goward, paragraph [0048].

To establish *prima facie* obviousness of a claimed invention, all the claim limitations must be taught or suggested by the prior art. *In re Ryoka,* 180 U.S.P.Q. 580 (C.C.P.A. 1974). *See also In re Wilson,* 165 U.S.P.Q. 494 (C.C.P.A. 1970). In the present case, Goward cited as teaching the features of claim 3 fails to do so because nowhere does Goward teach that a "user interface template contains HTML code". Removal of the pending rejection is respectfully requested and allowance is earnestly solicited.

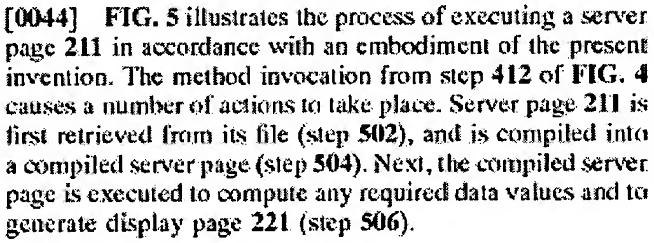
Claim 6 recites, "[o]ne or more computer-readable memories containing a computer program that is executable by a processor to perform the method recited in claim 1." Applicant respectfully submits that claim 6 is allowable for the same reasons as are discussed with respect to claim 1. Removal of the pending rejection is respectfully requested and allowance is solicited.

Claim 7, in part recites:

- identifying a plurality of user interface templates created using an Active Server Page Language and associated with a Web-based application;

- compiling each of the plurality of user interface templates into a single file containing a plurality of byte codes, wherein the byte codes are capable of being executed by an execution engine that implements an Internet service application programming interface (ISAPI) of the Web-based application; and
- executing the plurality of byte codes when the Web-based application is executed.

The pending rejection under 35 U.S.C. §103(a) over Goward in view of Viwanath is improper as the combination fails to teach or suggest "compiling each of the plurality of user interface templates into a single file containing a plurality of byte codes". For the Office's convenience, Goward, paragraph [0044] including step 504 contended by the Office as teaching this feature is reproduced directly below.



> [0044] **FIG. 5** illustrates the process of executing a server page **211** in accordance with an embodiment of the present invention. The method invocation from step **412** of **FIG. 4** causes a number of actions to take place. Server page **211** is first retrieved from its file (step **502**), and is compiled into a compiled server page (step **504**). Next, the compiled server page is executed to compute any required data values and to generate display page **221** (step **506**).

Goward, paragraph [0044].

Neither Goward step 504 nor anywhere else does Goward teach "compiling each of the plurality of user interface templates into a single file". In particular, Goward's templates are used to access the server pages. Thus, the templates in Goward are not "user interface templates" as recited in claim 7. Furthermore, while Goward step 504 teaches compiling a server page, this passage fails to teach "compiling each of the plurality of user interface templates into a single file" as is recited in claim 7.

The combination of Goward/Viswanath fails to teach the feature of "executing the plurality of byte codes when the Web-based application is executed" because the contended templates in Goward are used to access the server page rather than being used for "compiling each of the plurality of user interface templates into a single file containing a plurality of byte codes" in which the byte codes are executed "when the Web-based application is executed." For at least the foregoing reasons, the pending rejection under 35 U.S.C. §103(a) is improper. Removal of the pending rejection is respectfully requested and allowance is solicited.

Claims 8-11 depend from claim 7 and are allowable as depending from an allowable base claim. These claims are also allowable for their own recited features which are neither taught nor suggested by the combination of Goward/Viswanath.

Claim 12 recites, "[o]ne or more computer-readable memories containing a computer program that is executable by a processor to perform the method recited in claim 7." Applicant respectfully submits that claim 12 is allowable for the same reasons as are discussed with respect to claim 7. Removal of the pending rejection is respectfully requested and allowance is solicited.

Claim 13, in part recites:

- creating a plurality of user interface templates associated with a Web-based application, wherein the plurality of user interface templates are created using an Active Server Page Language and the Web-based application uses an Internet service application programming interface (ISAPI) to implement business logic separately from the plurality of user interface templates;
- compiling the plurality of user interface templates into a plurality of byte codes prior to execution; and
- storing the plurality of byte codes associated with the plurality of user interface templates in a single file, wherein the byte codes are capable of being executed by an execution engine in a Web server, the execution engine comprises run time code of the ISAPI that executes the single file derived from the plurality of user interface templates created using an Active Server Page Language to generate Web pages using a system language of the ISAPI.

The pending rejection over the combination of Goward/Viswanath is improper as nowhere does Goward teach or suggest "user interface templates." The Office's citation of Goward paragraph [0036] for this teaching is misplaced as Goward's templates are merely used to access a server page. While the Office additionally cites Viswanath paragraph [0064]-[0065] for this teaching, the Office's reliance is misplaced as this passage from Viswanath merely teaches compiling a JSP into a servlet rather than "compiling the plurality of user interface templates into a plurality of byte codes". Thus, Viswanath also fails to teach the feature of "the single file derived from the plurality of user interface templates created using an Active Server Page Language to generate Web pages using a system language of the ISAPI" because the compiled bytecode in Viswanath is compiled into a servlet rather than being used to generate "Web pages using a system language of the ISAPI."

As noted with respect to claim 1, the Office has failed to show that Viswanath teaches the feature of the "plurality of user interface templates created using an Active Server Page Language to generate Web pages". Absent some showing by the Office that Viswanath teaches this feature, the rejection is improper.

The combination of Goward/Viswanath also fails to teach the feature of "compiling the plurality of user interface templates into a plurality of byte codes prior to execution" as Goward step 504 is limited to teaching compiling the server page. This teaching in Goward fails to teach "compiling the plurality of user interface templates" as Goward's templates are used to access the server page and Goward's templates are external to the server page. Goward Figure 2. Therefore, Goward fails to teach compiling of "the plurality of user interface templates".

The combination of Goward/Viswanath also fails to teach "storing the plurality of byte codes associated with the plurality of user interface templates in a single file" because Goward never compiles the templates or stores the "plurality of byte codes associated with the plurality of user interface templates in a single file". Goward step 504 discussed in paragraph [0044] does not teach this feature as Goward teaches compiling the server pages.

For at least the above reasons, a *prima facie* case of obviousness under 35 U.S.C. §103(a) has not been shown. For at least the above reasons, removal of the pending rejection is respectfully requested and allowance is earnestly solicited.

Claims 14-16 depend from claim 13 and are allowable as depending from an allowable base claim. These claims are also allowable for their own recited features which are neither taught nor suggested by the combination of Goward/ Viswanath.

Claim 17 recites, "[o]ne or more computer-readable memories containing a computer program that is executable by a processor to perform the method recited in claim 13." Applicant respectfuliy submits that claim 17 is allowable for the same reasons as are discussed with respect to claim 13. Removal of the pending rejection is respectfully requested and allowance is solicited.

Claim 18 is allowable for at least the reasons discussed above with respect to claim 1. Applicant respectfully notes that the pending office action fails to address the particular language of claim 18. In particular, as claim 18 is directed to an apparatus (rather than a method) the pending office action fails to address the features recited in claim 18. No response is believed to be due as the Office has failed to present a *prima facie* case of obviousness under 35 U.S.C. §103(a) with regard to the features recited in claim 18. Applicant reserves the right to bolster/provide additional arguments. For at least the foregoing reasons, the pending rejection is improper. Removal of the pending rejection is respectfully requested and allowance is earnestly solicited.

Claims 19-20 depend from claim 18 and are allowable as depending from an allowable base claim. These claims are also allowable for their own recited features which are neither taught nor suggested by the combination of Goward/ Viswanath.

In particular, the combination of Goward/ Viswanath fails to teach the features recited in claim 19 because neither Goward nor Viswanath teach or suggest "the Active Server Page contains a plurality of byte codes associated with the plurality of user interface templates." Goward fails to teach this feature as Goward's templates are used to access server pages and are not included in the server pages. Goward, paragraph [0036] and Figure 2. Additionally, Viswanath is silent on the issue of "byte codes associated with the plurality of user interface templates" as the cited portions in Viswanath merely describe compiling JSP into bytecode, e.g., a servlet, that can be deployed to provide additional processing. Viswanath, paragraph [0065]. Removal of the pending rejection is respectfully requested and allowance is earnestly solicited.

Claim 21 is allowable for at least the reasons discussed above with respect to claim 7. Applicant respectfully notes that the pending office action fails to address the particular language of claim 21. In particular, the Office has failed to address the means features recited in claim 21 in accordance with M.P.E.P. §2181. As the pending office action fails to address the features recited in claim 21, no response is believed to be due. Applicant reserves the right to bolster/provide additional arguments. For at least the foregoing reasons, the pending rejection is improper. Removal of the pending rejection is respectfully requested and allowance is earnestly solicited.

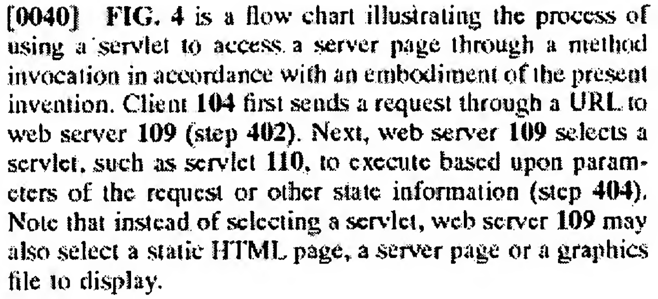
Claims 22 and 23 depend from claim 21 and are allowable as depending from an allowable base claim. These claims are also allowable for their own recited features which are neither taught nor suggested by the combination of Goward/Viswanath.

Claim 24, in part, recites:

> - create a plurality of user interface templates associated with a Web-based application, wherein the plurality of user interface templates are created using an Active Server Page Language and the Web-based application uses an application programming interface in a system language to implement business logic separately from the plurality of user interfaces;
> - compile the plurality of user interface templates into a plurality of byte codes of the system language; and
> - store the plurality of byte codes in a single file, wherein the byte codes are capable of being executed by a Web server that implements the application programming interface of the system language to generate Web pages.

The pending rejection of claim 24 is improper because Goward/Viswanath fail to teach each and every feature as recited in claim 24. For example, consider the feature of "create a plurality of user interface templates associated with a Web-based application", the combination of Goward/Viswanath fails to teach this feature as Goward paragraph [0040] describes the web server selecting a static

HTML page instead of selecting a servlet. Goward, paragraph [0040], reproduced below.



[0040] **FIG. 4** is a flow chart illustrating the process of using a servlet to access a server page through a method invocation in accordance with an embodiment of the present invention. Client **104** first sends a request through a URL to web server **109** (step **402**). Next, web server **109** selects a servlet, such as servlet **110**, to execute based upon parameters of the request or other state information (step **404**). Note that instead of selecting a servlet, web server **109** may also select a static HTML page, a server page or a graphics file to display.

Goward, paragraph [0040].

Thus, in contrast to the rest of the Office's rejection that relies on the use of a servlet, Goward paragraph [0040] specifically teaches that a servlet is not used. Moreover, the Office's pending rejection is improper as Goward fails to teach "user interface templates." Instead, Goward teaches templates 201-203 that are used for accessing server pages. Goward, paragraph [0036].

Additionally, the Goward/Viswanath combination fails to teach or suggest the feature of "compile the plurality of user interface templates into a plurality of byte codes of the system language". The Office's citation Viswanath paragraph [0064] for this teaching is misplaced as this passage is limited to compiling JSP bytecode into a servlet that can be used to access server pages. Therefore, the Goward/Viswanath combination would also fail to teach or suggest the feature of "wherein the byte codes are capable of being executed by a Web server that implements the application programming interface of the system language to generate Web pages" because the servlet is used to access server pages rather than

Web pages as recited in claim 24. For at least the above reasons, the pending rejection is improper. Removal of the pending rejection is respectfully requested and allowance is earnestly solicited.

Claims 25 and 26 depend from claim 24 and are allowable as depending from an allowable base claim. These claims are also allowable for their own recited features which are neither taught nor suggested by the combination of Goward/Viswanath.

Conclusion

The Application is in condition for allowance and the Applicant respectfully requests reconsideration and issuance of the present application. Should any issue remain that prevents immediate issuance of the application, the Examiner is requested to contact the undersigned attorney to discuss the unresolved issue.

Respectfully submitted,

Date: 8/28/08

By: /Nathan Grebasch/
Nathan Grebasch
Reg. No. 48600
Attorney for Applicant

Sadler, Breen, Morasch & Colby, PS
422 W. Riverside Avenue, Suite 424
Spokane, Washington 99201
Telephone: (509) 755-7267
Facsimile: (509) 755-7252